

INICIAR SESIÓN

NUESTROS PLANES

TODOS LOS
CURSOS

FORMACIONES

CURSOS

PARA
EMPRESAS

ARTÍCULOS DE TECNOLOGÍA > PROGRAMACIÓN

Entidades Managed, Transient y Detached en Hibernate y JPA



Paulo Silveira

21/07/2021

Distintuir entre los estados de una entidad en JPA / Hibernate es difícil al principio. Se dice que un objeto es transiente cuando no tiene representación en la base de datos y tampoco el EntityManager lo conoce, como a continuación:

```
Cliente c = new Cliente();
```

Aquí, cualquier cambio en el objeto referido por C no generará ningún tipo de insert o update en la base de datos. Lo contrario es cuando el objeto existe en la base de datos y el EntityManager en cuestión tiene una referencia a él, esta entidad está *managed*, gestionada por el EntityManager. Considera en una referencia a un EntityManager en el siguiente ejemplo:

```
Cliente c = new Cliente(); // transiente
```

```
em.persist(c); // gestionado
```

O también:

```
Cliente c = em.find(Cliente.class, 1); // gestionado
```

Cuando una entidad está *managed*, cualquier cambio en su estado (como una llamada de *setter*) resultará en una actualización de la base de datos en el momento del *commit*.

El último caso es cuando la entidad representa algo que posiblemente esté en la base de datos, pero el *EntityManager* lo desconoce: la entidad está fuera de contexto, *detached*. Ejemplo:

```
Cliente c = new Cliente();  
  
c.setId(1);
```

Una entidad también está *detached* cuando el *EntityManager* de donde obtuvimos este *Cliente* (por ejemplo, cuando hicimos un *find* o proveniente de una *Query*) ya no está abierta. Cualquier cambio en esta referencia obviamente no tendrá ningún efecto en la base de datos. Para que este cambio surta efecto, es decir, para que *reattach* a la entidad, primero debemos vincularla al contexto de persistencia. Tenga en cuenta que en el *EntityManager* puede que ya exista una entidad *Cliente* con esta misma *id*, imagina lo que sucedería si tuviéramos un método que se llamara *reattach* o *update*?

Entonces el método es el *merge*. Une la posible entidad con la misma *id* que está en el *EntityManager* con la informada como argumento, y devuelve la que está *managed*. El método *merge* **no** hace *reattach*. Luego:

```
Cliente c = new Cliente();  
  
c.setId(1);  
  
em.merge(c);  
  
c.setNome("Cliente con nombre cambiado");
```

¡No funcionará! Aquí deberías haber tomado lo que devolvió el *merge*. Nota el pequeño cambio:

```
Cliente c = new Cliente();  
  
c.setId(1);  
  
c = em.merge(c);  
  
c.setNome("Cliente con nombre cambiado");
```

Listo. Una pequeña introducción sobre el ciclo de vida de una entidad en relación con a un EntityManager: ¡*transient* (la especificación se llama *new*), *managed* y *detached*! También tenemos el estado *removed*, cuando una entidad está marcada para su eliminación.

Puedes leer también:

- [JPA con Hibernate: Herencia y Mapeos](#)
- [Entendiendo el Lazy y el Eager Load de JPA](#)
- [Conozca la API de fechas de Java 8](#)

ARTÍCULOS DE TECNOLOGÍA > PROGRAMACIÓN

**En Alura encontrarás variados cursos sobre Programación.
¡Comienza ahora!**

SEMESTRAL

US\$49,90

un solo pago de US\$49,90



- 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 6 meses

¡QUIERO EMPEZAR A ESTUDIAR!

[Paga en moneda local en los siguientes países](#)

ANUAL

US\$79,90

un solo pago de US\$79,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español

- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 12 meses

¡QUIERO EMPEZAR A ESTUDIAR!

[Paga en moneda local en los siguientes países](#)

Acceso a todos
los cursos

Estudia las 24 horas,
dónde y cuándo quieras

Nuevos cursos
cada semana

NAVEGACIÓN

PLANES

INSTRUCTORES

BLOG

POLÍTICA DE PRIVACIDAD

TÉRMINOS DE USO

SOBRE NOSOTROS

PREGUNTAS FRECUENTES

¡CONTÁCTANOS!

¡QUIERO ENTRAR EN CONTACTO!

BLOG

PROGRAMACIÓN

FRONT END

DATA SCIENCE

INNOVACIÓN Y GESTIÓN

DEVOPS

AOVS Sistemas de Informática S.A

CNPJ 05.555.382/0001-33

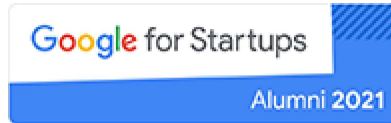
SÍGUENOS EN NUESTRAS REDES SOCIALES



ALIADOS



En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

CURSOS

Cursos de Programación

Lógica de Programación | Java

Cursos de Front End

HTML y CSS | JavaScript | React

Cursos de Data Science

Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

Cursos de DevOps

Docker | Linux

Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics | Liderazgo y Gestión de Equipos | Startups y Emprendimiento